Trust Issue(r)s: Certificate Revocation and Replacement Practices in the Wild

David Cerenius[1], Martin Kaller[1], Carl Magnus Bruhner[1], Martin Arlitt[2], and Niklas Carlsson[1]

[1] Linköping University, Linköping, Sweden
[2] University of Calgary, Calgary, Canada

Abstract. Every time we use the web, we place our trust in X.509 certificates binding public keys to domain identities. However, for these certificates to be trustworthy, proper issuance, management, and timely revocations (in cases of compromise or misuse) are required. While great efforts have been placed on ensuring trustworthiness in the issuance of new certificates, there has been a scarcity of empirical studies on revocation management. This study offers the first comprehensive analysis of certificate replacements (CRs) of revoked certificates. It provides a head-to-head comparison of the CRs where the replaced certificate was revoked versus not revoked. Leveraging two existing datasets with overlapping timelines, we create a combined dataset containing 1.5 million CRs that we use to unveil valuable insights into the effect of revocations on certificate management. Two key questions guide our research: (1) the influence of revocations on certificate replacement behavior and (2) the effectiveness of revocations in fulfilling their intended purpose. Our statistical analysis reveals significant variations in revocation rates, retention rates, and post-revocation usage, shedding light on differences in Certificate Authorities' (CAs) practices and subscribers' decisions. Notably, a substantial percentage of revoked certificates were either observed or estimated to be used after revocation, raising concerns about key-compromise instances. Finally, our findings highlight shortcomings in existing revocation protocols and practices, emphasizing the need for improvements. We discuss ongoing efforts and potential solutions to address these issues, offering valuable guidance for enhancing the security and integrity of web communications.

1 Introduction

Trust is paramount for secure communication on the web. To uphold the trust that we communicate with the correct service, HTTPS (HTTP over TLS), the dominant protocol for delivering web content, relies on X.509 certificates. At a high-level, each of these certificates binds a public key to the identity of the owner(s) of that key; this serves as a guarantee that the identity of an otherwise anonymous party (domain) is as claimed. However, maintaining the continued trustworthiness of these certificates requires not only their proper issuance and

This version of the contribution has been accepted for publication in Proceedings of Passive and Active Measurement (PAM) Conference 2024, after peer review (when applicable) but is not the Version of Record and does not reflect post-acceptance improvements, or any corrections. The Version of Record will be available through Springer Nature (DOI not yet available). Use of this Accepted Version is subject to the publisher's Accepted Manuscript terms of use https://www.springernature.com/gp/open-research/policies/accepted-manuscript-terms

management, but also their timely revocation in case of compromise, fraudulence, or misuse. Two essential aspects of maintaining the integrity and security of web communications are therefore to (1) revoke certificate that are no longer trusted and (2) replace them with new, trustworthy certificates.

In this paper, we present the first data-driven characterization of the certificate replacements (CRs) of revoked certificates, in which we provide a head-to-head comparison of the CRs associated with revoked vs. non-revoked certificates. By taking advantage of two existing and complementing datasets with overlapping collection periods, we first create a combined dataset consisting of 1.5 million CRs and their associated revocation statuses. Using this dataset, we then compare the CRs for which the replaced certificate was revoked versus those for which it was not revoked and provide new insights into the effects that revocations have on certificate management. To help guide the research, we aimed to address two previously not answered questions:

- What effect(s) do revocations have on certificate replacement behavior?
- How effectively are replacements preventing post-revocation usage?

In our analysis, we use statistical comparisons of several properties to identify differences in the characteristics of revoked and non-revoked certificates. Our study reveals statistically significant discrepancies that can be attributed to differences in the certificate management practices seen across the issuing Certificate Authorities (CAs). For example, the revocation rates vary from a fraction of a percent to over 17% for individual CAs that partially can be mapped to differences in their revocation requirements. We also found that the retention rates varied significantly among CAs, indicating that the CA's handling of revocations and overall satisfaction with their services affect subscribers' decisions.

In addition, a notable percentage of certificates were either directly observed to have been used (meaning actively provided as part of an HTTPS handshake, collected in one of our datasets) after revocation or estimated to have been used after revocation. For example, 7% of the revoked certificates were observed after their revocations when preferably a reissued or otherwise replacing certificate should be used instead. Using the replacing certificate as an indicator of a certificate's actual lifetime, we estimated that at least 24% of the revoked certificates in our dataset were used despite their revoked status, with the periods of illegitimate advertisement ranging from a few days to multiple years. The extent of post-revocation usage varied by CA, validation types, and revocation reasons. Perhaps most concerning was that certificates with "Key Compromised" as the revocation reason had the highest observed post-revocation usage, raising concerns about how these possible key-compromise instances were handled.

Finally, to provide concrete guidance, we also use our findings and insights to highlight some of the current problems with the existing revocation protocols and practices, we also provide a discussion of ongoing efforts and possible solutions to address some of these issues.

The remainder of the paper is organized as follows. Section 2 introduces the necessary background on X.509 certificates and current revocation practices. Section 3 describes the datasets and the methodology used to combine the datasets.

In Section 4 we present our results, aimed at addressing the above guiding questions. We then discuss our findings, highlight problems with existing revocation protocols and practices, and discuss possible improvements in Section 5. Finally, Section 6 presents related work, before Section 7 concludes the paper.

2 Certificates and their lifetime

X.509 certificates: The X.509 standard defines the public key certificates used by TLS (and hence also HTTPS) to verify the legitimacy of the public keys used as part of the Public-Key Infrastructure (PKI) specified in RFC 5280 [4]. This RFC covers (among other things) certificate format, semantics, and standardized fields like serial numbers, issuer details, signatures, and validity periods, along with various extensions.

Certificate issuer: An X.509 certificate may either be self-signed or issued by a certificate authority (CA). As the term suggests, self-signed certificates can be generated by anyone as they are issued, received, and signed by the same entity (for instance a web domain) without third-party involvement [38,47]. Consequently, all major web browsers will reject these certificates as there is no way to validate them [38]. The more common approach is to utilize CAs, where a CA is a third-party entity/organization that issues, signs, and acts as a guarantor for the certificate's validity. CAs are also able to issue certificates to themselves in which case they are regarded as self-issued [4].

Validation type: While not part of the X.509 PKI specification, the standard-defining CA/Browser Forum defines three types of SSL/TLS certificates [7,9]:

- **Domain Validation (DV):** Issued as soon as ownership of the domain in question has been demonstrated (usually through e-mail validation).
- **Organization Validation (OV):** In addition to the domain, the issuing CA validates other information to verify the legitimacy of the organization.
- **Extended Validation (EV):** Similar requirements to OV certificates but with a much stricter vetting process (generally requiring more time).

A certificate in one of these categories is identified by an Object Identifier (OID) of the corresponding type in the `certificatePolicies` extension [4,9].

Validity period: The validity period of a certificate is the time span between the `notBefore` and `notAfter` timestamps, inclusive, defined in the certificate [4]. A CA is only required to maintain information about a certificate's status during this time. The set validity period varies depending on the CA and validation type. However, the general trend across CAs is that validity periods are getting shorter. For example, over the last few years the CA/Browser Forum Baseline Requirements (BR) [9] has gone from capping validity for new certificates at 39 months (∼1186 days) in 2015 to 398 days in 2020 [6], with intentions in 2023 to further reduce down to 90 days in the future [45].

Certificate replacements: To ensure that the certificates used by a domain (or web server) are valid and up-to-date, certificates must regularly be replaced. When a certificate is replaced by another certificate, a certificate replacement

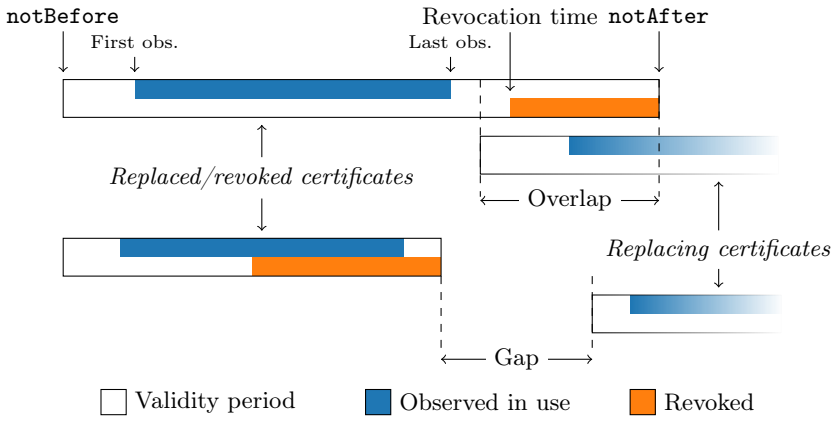


Fig. 1. Validity overlap and gap. Definitions are the same with or without a revocation.

(CR) occurs. While replacements generally happen near the end of a certificate's validity period [5], some replacements are done with some margin ahead of the certificate expiry while others are made after expiry. We define a validity *overlap* as the intersection of the new certificate's validity period with that of the previous one. Similarly, we say that a validity *gap* occurs (see Figure 1) if there is a discontinuity in the two validity periods. Having validity overlap is a good practice (often simplified and facilitated by automation) that is important to guarantee availability of a service, since a validity gap will result in an invalid period impacting service access (serving as a mismanagement indicator [5]).

For the analysis in this paper, we use the CRs identified by Bruhner et al. [5]. Focusing on certificates observed in use via large-scale scans, they define a CR as a relation between a pair of observed certificates, where the IP address, port number, and `subjectCN` (entity name, including cases of so-called wildcard subdomains) of the certificates match, thus capturing them being valid for the same entity and usage. Additionally, to form a CR, the replacing certificate must begin and end its validity period later than the begin and end, respectively, of the replaced certificate's validity period, allowing for both overlaps and gaps as defined above. For full details, we refer to the paper by Bruhner et al. [5].

Certificate revocations: There are several reasons that the trust in a certificate must be revoked before it expires. For example, if a certificate's private key is compromised or an integral information field warrants modification, the issuing CA might be obliged to revoke (invalidate) the certificate in question [4]. There are currently two main methods of revocation (both having some shortcomings and later improvements that we will discuss in 5):

– **Certificate revocation list (CRL):** As per the original X.509 PKI certificate RFC [4], a CRL issuer (typically a CA) publishes lists of revoked certificates, available through a link under the `cRLDistributionPoints` extension of the certificate. This link can be used to download the CRL and ascertain the presence of the serial number associated with a given certificate.

– **Online Certificate Status Protocol (OCSP):** Within the X.509 PKI, OCSP, defined in a separate RFC [37], provides real-time revocation information on an on-request basis [37]. In contrast to CRLs, updated every 24 hours by CAs, OCSP servers always deliver up-to-date revocation details without the need to fetch an entire list of serial numbers (which might contain several hundred thousand entries), and certificate statuses are generally removed within seven days of certificate expiry [23].

Certificate Transparency (CT): Since 2018, both Google and Apple [10] require CAs to add all issued certificates (regardless of type) to one or more append-only logs [26]. The main goal of CT is to combat misissuance of certificates by allowing the public to audit them. The logs can be operated by anyone as long as one follows the specified standards; however, the largest ones are almost exclusively run by CAs or browser vendors [40]. As of today, there is no widely implemented revocation transparency solution. Instead, CRLs (when available) are the most reliable source for revocation times and revocation reasons.

3 Dataset creation

This section describes the originating datasets and how they were merged and refined to create a combined dataset that fits the purpose of our analysis.

3.1 Original datasets

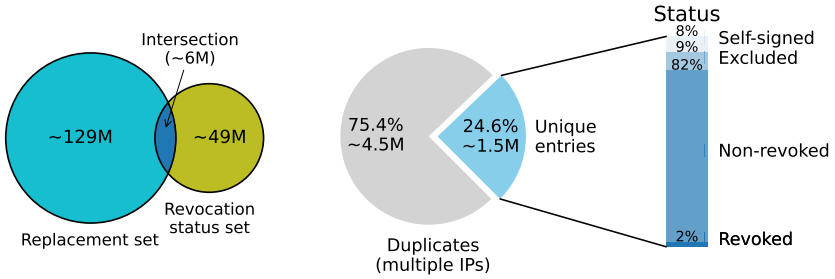
This paper leverages complementing features of two existing datasets with overlapping collection periods. The first dataset contains a large number of observed *certificate replacements* [5], capturing the replacing certificate and relative timing of when a certificate was replaced. The second dataset contains all *certificate revocations* with certificates that were revoked before their expiry and that had an expiry date between March 2 and April 1, 2020 [23].

Replacement dataset [5]: The replacement set was derived from publicly available Rapid7 logs collected as part of Project Sonar [35,36]. In particular, the certificates observed in weekly/biweekly scans of port 443 across the full IPv4 address space were used to reconstruct (artificial) certificate replacement (CR) relations. This dataset spans from Oct. 2013 to July 2020, and contains 129 million CRs.

Revocation status dataset [23]: The revocation status set was compiled using Certificate Transparency (CT) logs to check for any certificates expiring within the period of March 2 to April 1, 2020. The day before expiry (and periodically for 120 days afterwards in case of revocation) OCSP statuses were collected and saved for every certificate. In total, this dataset contains 49 million certificates out of which 1.08 million (2.18%) were revoked.

3.2 Creation of a combined dataset

For our analysis, we created a combined dataset that includes both certificate replacement and revocation information for a large set of certificates expiring



(a) Datasets' intersection visualized (b) Overlap data visualized

Fig. 2. Overview of the dataset analyzed in this paper.

between March 2 and April 1 (2020). This is made possible by recognizing that the CRs collected by Bruhner et al. [5] includes the CRs for a large set of the 49 million certificates that Korzhitskii and Carlsson [23] collected revocation status information about (all with expiry during the period above). First, we considered all CRs (from the set of 129 million CRs) for which the replaced certificate expired between March 2 and April 1, 2020 (and for which we therefore had access to accurate revocation information in the 49 million certificates of the revocation status set). This corresponded to 6 million CRs (as illustrated in Figure 2a), to which we appended the data of the revocation status set.

Next, we removed all duplicate CRs (i.e., those that differed only in the IP addresses where the certificates were observed), after which we were left with 1.5 million certificates (as illustrated in Figure 2b). We then removed self-signed certificates (as those are not included in the revocation status set)—noting that these certificates have been found to make up the majority of invalid certificates on the internet [11]—as well as all certificates for which the OCSP queries did not offer either a status `revoked` or `good` (we denote these "Excluded", also including a mix of timeouts as well as the `unknown` and `unauthorized` responses). With this, we were left with 1.2 million unique CRs for our combined dataset.

3.3 Augmenting with revocation times and revocation reasons

As the revocation status set did not provide revocation times or reasons, we augmented the dataset by collecting more than 300,000 complete CRLs from the roughly 2,400 distribution points (gathered by Korzhitskii et al. [23]).

CRLs usually contain multitudes of revoked serial numbers and their associated revocation times/reasons. This posed a bit of a challenge as there was no mapping between the CRL distribution point URL and CA. This led to every serial number having to be individually searched for among the approximately 250 GB of CRLs available to us. In total, the revocation times and reasons for 3,768 certificates ($\approx$ 10% of all revocations, as explained below) were successfully obtained and are used for the analysis of revocation time in Section 4.7. Notably, this does not include data for Let's Encrypt, constituting 89% of all revocations,

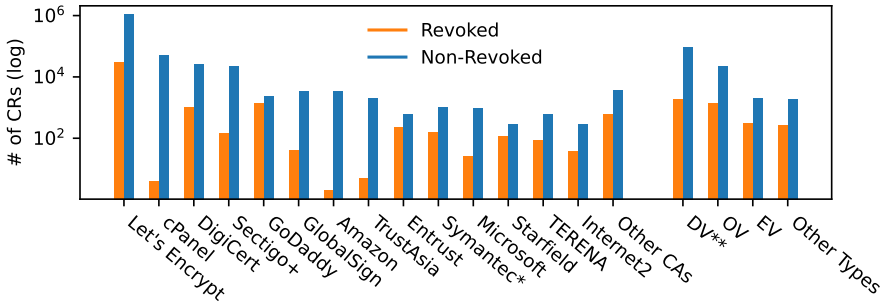


Fig. 3. Total number (in logarithmic scale) of CRs per CA and type.

as they did not support CRL until September 2022 [17], and TrustAsia that did not operate their own root or CRL before August 2020 [46].

3.4 Limitations

Neither of the two used datasets are perfect. First, the replacement set contains only certificates found via IPv4 scans of port 443. While broader scans (e.g., more ports) could have increased the observed lifetimes, we note that most HTTPS servers use port 443. Second, the Project Sonar scans are conducted relatively infrequently (biweekly) and occasionally miss certificates. Since this affects the granularity with which we can estimate the observed lifetimes of certificates, we report both observed and approximated lifetime values, the latter being a calculated approximation explained in Section 4.7. Third, the scans may also miss many certificates found in CT logs, potentially causing us to miss some certificates of interest. However, this dataset has the advantage that it captured certificates in use (not only certificates with some intended validity period), helping us focus on certificates observed in the wild. Fourth, the dataset is from 2020. As noted, several changes in the certificate landscape have since happened and, at that time, Let's Encrypt did not use CRLs, preventing timing analysis of their certificate revocations (possible for most other revoked certificates).

3.5 Ethical statement

This paper does not pose any ethical issues. By using datasets collected for prior works, we limit the extra load faced by servers. Furthermore, the measurements used here (e.g., IP scans, OCSP status requests, and CRLs) are expected to contribute only a small portion of the overall load typically seen by the servers. Finally, all data were collected from public infrastructures using public protocols.

4 Characterization Results

4.1 Summary of dataset

Figure 3 shows the total number of revoked and non-revoked CRs for all CAs with at least 500 CRs with roots approved certificates. Here, we say that a certificate

Table 1. Selected certificate authorities sorted by number of CRs.

Issuing CA	# of CRs	Revocation rate (%)	Validation type (%)		
			DV	OV	EV
Let's Encrypt	1,106,587	2.75	100	0	0
cPanel	98,819	0.01	100	0	0
DigiCert	35,299	2.92	22.69	73.35	3.91
Sectigo+	28,663	0.50	94.39	4.48	1.13
GoDaddy	7,756	17.07	96.29	2.15	1.13
GlobalSign	4,916	0.81	41.33	56.79	1.77
Amazon	3,259	0.06	100	0	0
TrustAsia	2,080	0.24	99.90	0.10	0
Entrust	1,640	13.48	0	83.11	16.89
Symantec*	1,188	12.96	56.90	42.93	0
Microsoft	1,030	2.43	0	0	0
Starfield	888	12.61	97.07	1.91	1.01
TERENA	705	11.77	0	76.45	23.55
Internet2	652	5.83	0	100	0
Other CAs	59,051	1.01	1.66	2.71	0.19

is root approved if it was validated by all three major browser vendors' trust stores at the time of the collection (Apple, Microsoft, and Mozilla/NSS; Chrome had not yet released its own root store at the time [44]). As a reference point, we also included an "Other" category that contained all other CRs. To simplify the reading going forward, Sectigo, including certificates issued under their old Comodo brand, will be denoted as *Sectigo+*; Symantec, including certificates issued by GeoTrust (when they were still under the Symantec PKI), will be denoted as *Symantec**; and DV certificates, excluding certificates issued by Let's Encrypt (as explained in the next section), will be denoted *DV***.

We note that Let's Encrypt dominates the dataset, accounting for approximately 83% of the total number of CRs and 89% of all revocations. No other CA comes close in terms of pure quantity. Part of the high Let's Encrypt numbers are due to their 90-day validity policy and widespread use of their ACME client (Certbot) that simplify both replacement and revocations, but also a mass-revocation event that took place around the time of the data collection.

Note: To simplify reading, we use the name of the CA to refer to certificates issued by that CA. Nevertheless, we stress that it is ultimately the subjects that are responsible for the certificate's usage, including many types of revocations.

4.2 Revocation rates

Table 1 summarizes the revocation rates of certificates of all root approved CAs with at least 500 CRs. To simplify comparisons of CAs, we also include a breakdown of each CA's share of DV, OV, and EV certificates observed in the CRs.

Big differences in CAs' revocation rates: While the overall revocation rates are similar to those observed in prior works (e.g., [19,23,29,48]), and

revocations typically are relatively rare (all things considered), we observe big differences between the revocation rates of individual CAs, with revocation rates ranging from a fraction of a percent (Amazon and cPanel) to 17% (GoDaddy).

At this time, it should also be noted that part of the higher-than-average revocation rate seen for Let's Encrypt (2.75%) is due to a mass revocation event that took place in March 2020. This event was triggered by a Certification Authority Authorization (CAA) rechecking bug [28] (when the revocation data was collected) but did not affect the security of the users. As Let's Encrypt only offers DV certificates, this results in DV certificates being the primary validation type. For this reason, we excluded Let's Encrypt from the DV category when presenting data as they would otherwise essentially be identical.

From the table, it is also apparent that most of the CAs with fewer CRs (Starfield, Symantec, Internet2, and TERENA) have noticeably higher revocation rates in general compared to the larger CAs (e.g., Let's Encrypt or DigiCert). This is consistent with a similar study on revoked certificates, retrieved from CT Logs, by Halim et al. [19]. This phenomenon could in part be explained by smaller CAs having different customer bases (the same study found for example that a majority of Starfield's revocations was due to domain owners ceasing operations). While this pattern does not match the lower revocation rates observed for the "Other" category, we note that this category also contains certificates that are not validated by the major root stores and other certificates that for which we saw more irregular behaviors.

Influence of revocation policies: Another factor that appears to affect the relative revocation rates seen with different CAs is their revocation policies. While all CAs are expected to follow the CA/Browser Forum Baseline Requirements (BRs) [9], we have found (in a not yet published related work) that there are differences in who can request revocations, revocation procedures, and revocation reasons. Of special note here is Amazon, who has the second-lowest revocation rates of the reported CAs but also has (1) the most stringent requirements for who can request revocation (exclusively the subscriber) and (2) the most unaccommodating revocation procedures (no round-the-clock support for Certificate Problem Requests and no publicly readily available revocation instructions). At the other end of the spectrum, we have found that Starfield states that they will revoke a certificate if the subscriber fails to pay any invoice, possibly contributing to their higher-than-average revocation rates.

4.3 Issuer changes and CA retention rates

The decision of whether a subscriber chooses to remain with their current CA or to switch to a new one when replacing their certificate can reveal a lot about revocation procedures and CA satisfaction. To investigate to what degree customers are more or less likely to change CAs when revoking their certificate, we next look at the percentage of CRs where the replacing certificate was issued by a different CA than the CA that issued the replaced certificate of a CA. Figure 4 summarizes these results.

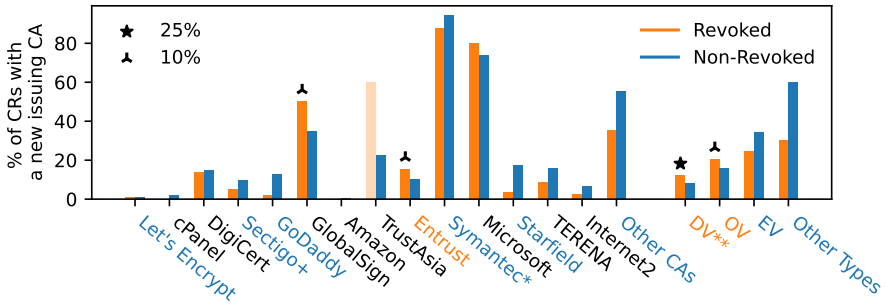


Fig. 4. Percentage of CRs with a different issuing CA for the replacing certificate. A transparent bar/box denotes five or fewer datapoints. The black symbols signify the degree of Z-test bias for the revoked population and a colored CA/type-name indicates a Fisher bias in the respective direction that the color represents.

Statistical tests and visual annotations: For these results and some of the later results, we used different forms of statistical testing to reveal statistically significant biases and enhance the data representation in certain graphs. As far as sample sizes allowed, we used single-tailed, two-proportion Z-tests to compare revoked ratios with non-revoked (times alternating constants, $(1.1/1.25/2)$). When an insufficient number of data points were available for those tests, we applied binomial tests as a complement, with n and k being taken from the revoked population and the "true" proportion p coming from the non-revoked population. Both of these tests use a significance level of 90% and will be presented as a range of symbols in the bar charts depending on the degree of bias. This was also coupled with Fisher's exact test at a 95% significance level as a definitive indicator of bias between revoked and non-revoked samples.

Big differences in retention rates: We see a massive spread in the joint averages between CAs as Let's Encrypt, cPanel, and Amazon seem to have virtually no subscribers leaving them while Microsoft and Symantec both have noticeably lower retention rates. Symantec's case can, just as Bruhner et al. [5] theorized, be explained by Google over this time period implementing a plan to distrust all Symantec issued certificates [18].

Mostly small effect of revocations: At an aggregate level, we observe that there is not much of a difference between the revoked and non-revoked ratios. If anything, there is even a small bias towards the non-revoked CRs switching their issuing CA. The deviating CAs are GlobalSign and Entrust who both display a slight partiality of 10 percent in the revoked direction (even though GlobalSign is missing the Fisher bias). As for the types, revoked DV certificates having poorer CA retention rates could simply be explained by low validation requirements leading to an easy switch in case of subscriber dissatisfaction.

4.4 Validity

Figure 5 shows a box-and-whisker plot of the validity period of revoked certificates compared with non-revoked (divided by CA and type). While the validity

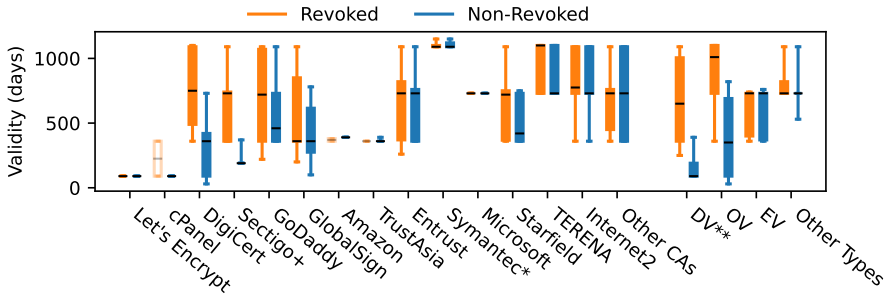


Fig. 5. Validity time measured in days per CA and type. The plotlines in this box-and-whisker plot are capped at the 10th and 90th percentile, with the box representing the 25th to 75th percentile, and a black line marking the median.

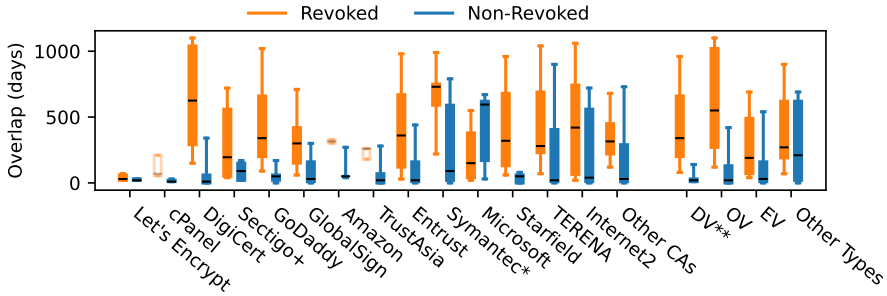


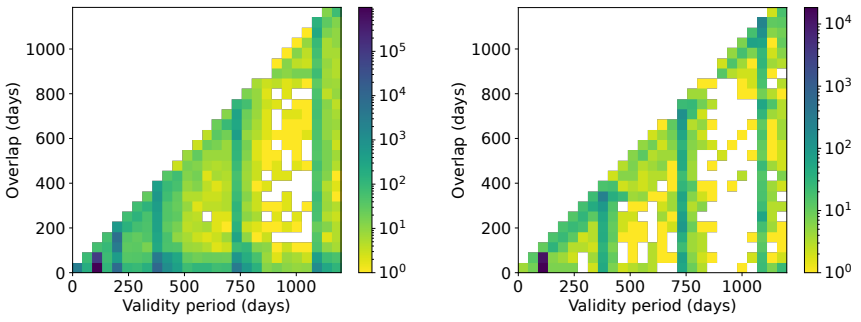
Fig. 6. Overlap in validity time for CRs measured in days, per CA and type.

periods vary greatly among CAs, some offer a diverse range of validity periods, while others (e.g., Let's Encrypt) offer only one. We note that for the CAs with variable validity periods, the revoked certificates typically had significantly longer validity periods than the non-revoked. For cPanel, two out of the four revoked certificates (transparent since too few samples) had a validity period of a year but were revoked immediately, suggesting they may have been misissued.

It is also worth noting that the revoked certificates in our dataset were issued before validity periods were capped at 398 days in 2020. The trend is particularly evident when examining larger CAs such as DigiCert, Sectigo, or GoDaddy as well as DV and OV type certificates. EV certificates breaking this tendency could be explained by the rigorous validation requirements leading to a more homogeneous subscriber group, especially compared to that of DV certificates.

4.5 Overlap

The degree of overlap between the two certificates in a CR captures to what extent the subscriber (or the CA in case of automated replacements, e.g., with Let's Encrypt and cPanel) desires a safety margin. Figure 6 compares the overlaps of revoked and non-revoked certificates. As perhaps expected, with the exception of Microsoft, the revoked certificates exhibit significantly higher levels of overlap



(a) Non-revoked (b) Revoked

Fig. 7. Heat map of the relationship between validity and overlap. Heat colors are represented on a logarithmic scale, with white indicating an absence of data.

compared to their non-revoked counterparts. This trend is primarily due to revocations frequently happening at the start of a certificate's validity period [19].

We expect that part of the reason for Microsoft diverging is that most of their non-revoked certificates have a (seemingly) static 2-year validity period and a median overlap of 600 days, meaning that most of their `good` certificates were only used for a few months before being replaced without getting revoked afterwards (considering Microsoft's low (2.43%) revocation rate).

Of the other CAs, some CAs stand out more than others when it comes to divergence in overlap for revoked vs. non-revoked CRs, with DigiCert and Symantec being particularly noteworthy. As for validation types, both DV and OV certificates also display big differences.

Implication of validity period: To better visualize the relationship between overlap and validity, we include Figure 7 which shows heat maps with these statistics for both non-revoked and revoked CRs, respectively. Per definition (and as seen in the figure) it is impossible for a CR to have a larger overlap than the validity period of any of the certificates (here that of the replaced certificate). As expected, we observe distinct heat zones within the validity periods of 90 days, 180 days, 1 year, 2 years, and 3 years for both categories of CRs. These were common options for validity offered by CAs at the time.

Early revocations and late replacements of non-revoked certificates: Comparing the figures more closely, we note that non-revoked CRs exhibit strong heat patterns at the bottom along the x-axis, indicating minimal overlap, whereas the revoked CRs show a more prominent middle diagonal (especially for certificates with longer validity) but minimal heat at the bottom. The strong middle diagonal signifies very early replacements (implying early revocations), matching early revocation patterns previously reported by Halim et al. [19], for example. As a corollary to this, we note the lack of instances (heat) at the bottom along the x-axis, which is reasonable considering that a revocation rarely happens at the very end of a certificate's validity as making revocations that late would serve little to no practical purpose.

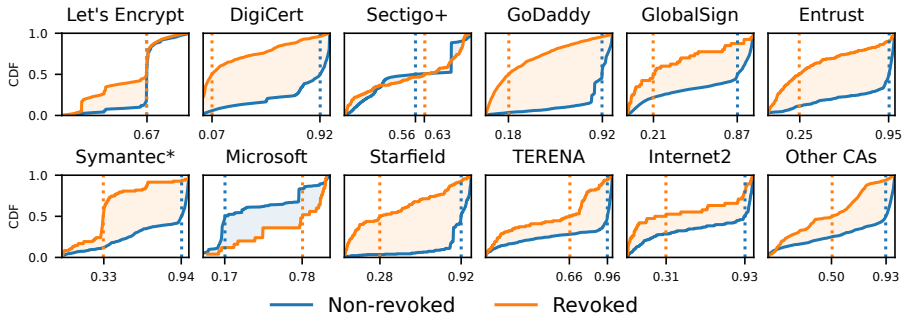


Fig. 8. Normalized CDFs of how far into the replaced certificate's validity overlap begins per CA. The dotted lines mark 50 percent of the respective population. cPanel, Amazon, and TrustAsia have been excluded due to lack of data.

Per-CA analysis: The CDFs presented in Figure 8 further corroborate our findings that revoked CRs exhibit earlier overlap than non-revoked ones. Here, for each CA, we show CDFs of the normalized overlap (i.e., overlap divided by validity period) for both revoked and non-revoked CRs. As we noted in Figure 6, DigiCert and Symantec (together with GoDaddy, Entrust, and Starfield) show the greatest disparity. Microsoft once again stands out as the sole CA with a significantly earlier overlap for non-revoked CRs than revoked ones. This supports the theory of how they only use their `good` certificates for a short while before replacing them. Specifically, observing the dotted lines in the CDF, half of Microsoft's non-revoked certificates have overlap just 17% into their validity period while the revoked ones reach the 50 percent mark at 78% into their validity. Also contrasting the rest, Let's Encrypt and Sectigo show very little difference between revoked and non-revoked CRs, which could potentially be an indicator of certificate misuse. We have already established that a revocation generally occurs early on in a certificate's lifetime, which should mean that revoked certificates also get replaced early on (the case for most of our CAs). This indicates that revoked CRs, like non-revoked ones with late replacements, may imply subscribers disregarding their certificate's revoked status.

Mass revocation event: Given that all certificates in our dataset expire between March 2nd and April 1st, coinciding with Let's Encrypt's mass revocation event in early March, it is of high interest to scrutinize the conduct of Let's Encrypt certificates during this period. Interestingly, most of their subscribers with revoked certificates still used the default automated renewal period [27] after two-thirds of the validity period has passed (orange CDF), something we see is even more rigorously followed for non-revoked certificates (blue CDF). Based on this, we can draw the conclusion that most certificates revoked in the event were already scheduled for replacement anyway, meaning the revocation had minimal effect on the replacement behavior of these certificates. Furthermore, given that 96.6% of our revoked Let's Encrypt certificates were part of the event, it is not surprising that the revoked and non-revoked CDFs are relatively homogeneous.

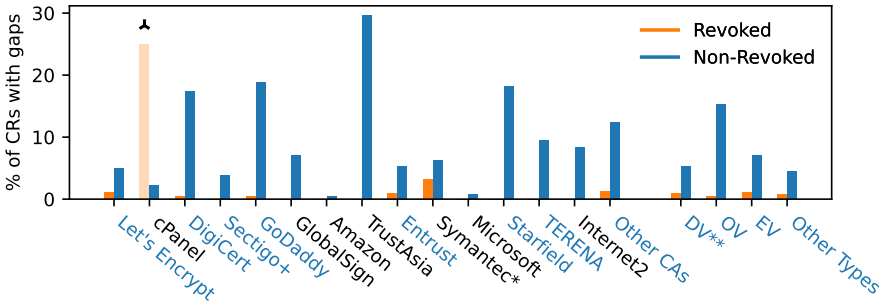


Fig. 9. Percentage of CRs with gaps in validity time per CA and type.

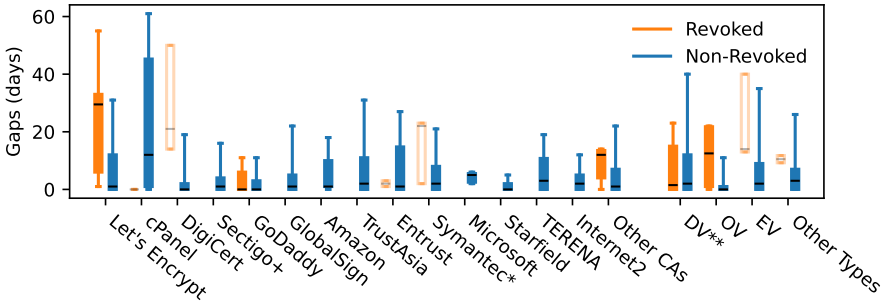


Fig. 10. Gaps measured in days per CA and type.

4.6 Gaps

Gaps are much more frequently observed in non-revoked compared to revoked CRs. This is shown in Figure 9, where we show statistics for the top CAs and the different revocation types. It is worth noting that DigiCert, GoDaddy, and Starfield exhibit an especially high rate of gaps among their certificates. The visibly obvious exception to this trend is cPanel. However, as seen by the transparent color of the revoked bar for cPanel, there is not enough data to draw any conclusions for cPanel. In particular, cPanel only has four revoked certificates (out of which one has a gap).

While it is rare for replacements with revoked certificates to contain gaps, it seems that when a gap does exist, it tends to be larger than in non-revoked CRs. This is illustrated in Figure 10, where we show the distribution of the gapped CRs. Interestingly, Let's Encrypt demonstrates the most substantial disparity between the two groups, which could be a mismanagement indicator for revoked certificates. As we have already established, revocations typically occur early in a certificate's validity period, which affords the subscriber ample time to obtain a replacement. Therefore, if a revoked certificate is replaced too late (resulting in a gap) it is more likely to be forgotten for a longer time period compared with non-revoked ones, as gaps in those CRs typically arise from the subscriber missing the expiry date. This always results in warnings to users that tend to result in the issue quickly being resolved.

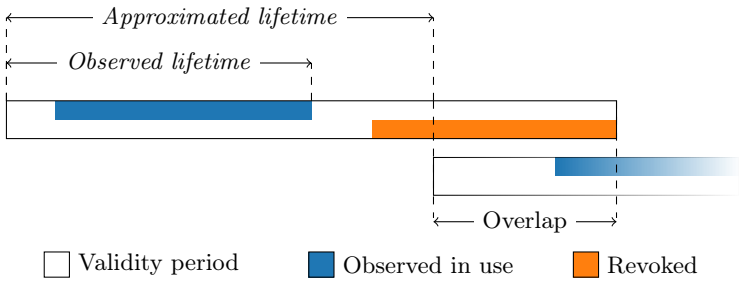


Fig. 11. Revoked certificates, with possible discrepancy between observed and approximated lifetime illustrated.

4.7 Revocation time analysis

To understand how quickly revoked certificates were replaced in practice, we compared the revocation times with the period over which each revoked certificate was used, which we call its *lifetime*.

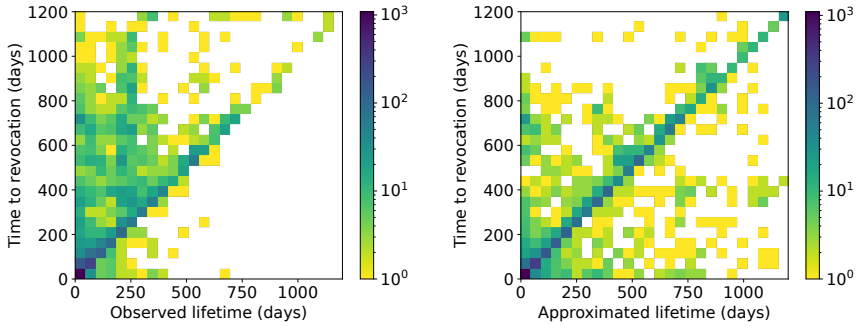
Revocation times: Using the CRL data, we were able to extract revocation times for every revoked certificate in our dataset except for those issued by Let's Encrypt and TrustAsia as explained in Section 3.3. This resulted in a set of 3,768 revoked certificates ($\approx$ 10% of all revocations) with recorded revocation times and reasons. We next describe two lifetime estimates.

Lifetime estimation: The time period between the beginning of a certificate's validity and the last time the certificate was observed in the Project Sonar scans is what we from hereon will call its *observed lifetime*. The observed lifetime is a conservative estimate of the actual lifetime due to the limitations in the originating scans in terms of frequency and responses, which gives a considerable error margin. To improve the potentially late discovery of new certificates, we use a certificate's `notBefore` date instead of the first observation since our main focus is to compare the end of the lifetime against the revocation time, meaning an accurate beginning of lifetime does not affect the comparison.

We also introduce a second way to calculate the lifetime of a certificate. We call this the *approximated lifetime*. This is the time between the replaced and replacing certificates' `notBefore` timestamps (as can be noted in Figure 1). In the majority of cases, the replacing certificate is at its earliest advertised on its `notBefore` date [11], meaning the replaced certificate was in all likelihood used up to that point. As a result, the approximated lifetime should be a better but still conservative estimate of the certificate's actual lifetime.

The problem of observation inaccuracy is demonstrated in Figure 11 where the revocation, in this case, takes place after the last observation making it seem as if the certificate never was illegitimately used. However, taking the replacing certificate into account makes it clear that the original certificate was indeed used despite being revoked.

Revocations in relation to their lifetime estimates: To capture this relationship, Figure 12 shows heat maps of (a) the observed lifetime against the revocation time and (b) the approximated lifetime against the revocation time.



(a) Observed (b) Approximated

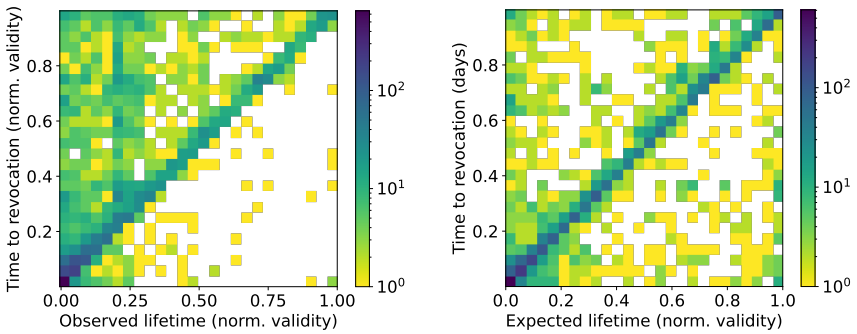
Fig. 12. Time to revocation and measured lifetime (days).

First, there is noticeable diagonal observed at $x = y$, where the revocation time and lifetimes are equal. In the observed lifetimes (Figure 12a) the majority of certificates are, however, located above this diagonal, meaning that the time to revocation is greater than the observed lifetime of the certificate. This implies that they were revoked sometime after the end of their lifetime. Still, there are some certificates under the $x = y$ diagonal, meaning those were observed to be used after being revoked. Specifically, out of the 3,768 revoked certificates, 281 were observed to be used after they were revoked, including 256 for over a day, 170 for more than a week, and 90 for more than a month after revocation.

Examining the approximated lifetime (Figure 12b), we observe a noticeable shift, indicating that many more certificates were likely used after revocation. Here, the $x = y$ diagonal is also more distinct, implying that a greater number of certificates were replaced in close proximity to their revocation times. Out of the 3,768 revoked certificates, 941 were approximated to be used after revocation with a threshold of at least one hour to increase accuracy. Among those, 824 were approximated to be used for more than a day, 500 for more than a week, and 311 for more than a month after revocation.

Our initial expectation was that most certificates would have a similar end-of-life time and revocation time, with the replaced certificate being revoked shortly after its subsequent replacement. However, a large portion of the certificates have an observed lifetime of less than 200 days while the time to revocation exceeds 200, 400, or even 600 days. When considering the approximated lifetime on the other hand, this pattern of revoking the certificate months/years after its replacement becomes much less pronounced and the diagonal becomes more prominent instead. It is also worth noticing that a great number of the certificates were observed only once, which might be the reason we see this behavior of late revocations in relation to the lifetime. With this in mind, our belief that the approximated lifetime gives a better estimate of the actual lifetime compared with the observations is strengthened.

Here, it should be noted that also the approximated lifetime sometimes underestimates the lifetime. For example, out of the 281 certificates observed after



(a) Observed (b) Approximated

Fig. 13. Time to revocation and measured lifetime (normalized).

revocation, 78 had a replacement certificate that could have been used instead. In those cases, despite that there appear to have been a new certificate issued that eventually were used, the subscriber kept advertising the revoked certificate for some time period after the revocation, leaving the valid replacement idle, before finally switching. For these cases, both lifetime estimates are underestimating the actual lifetime of the replaced certificate.

Normalized revocation-to-lifetime comparisons: To allow capture of the relative timings, Figure 13 shows a normalized heat map, where we normalize both the lifetime estimates and the revocation periods with the validity periods of the revoked certificates. Comparing with the non-normalized versions, we observe that the diagonal line now becomes more prominent using both lifetime measurements. This is due to the shorter validity certificates contributing along the whole line instead of just at the bottom. A phenomenon which also becomes more apparent is that there are quite a few certificates in the approximated lifetime graph (Figure 12b) that have had a lifetime equal to their validity yet were revoked very early on. As it seems quite unlikely for the subscriber to revoke their own certificate and then did not try to replace it, two probable explanations are that these certificates (i.e., that still are used after revocation) were involuntarily revoked, meaning that the subscriber might not have been aware of the event, or that the subscriber somehow failed to replace them (at least on the observed machines).

Post-revocation usage: To summarize the above results, 7.2% of the certificates have been observed while 24% have been approximated to be used after revocation (observed rate and approximated rate, respectively). When dividing these numbers into different categories however, such as issuing CA, validation types, or revocation reasons (see Figure 14), significant individual variation is revealed. Specifically, Microsoft had no observed revocation usage, while Sectigo, GoDaddy, and GlobalSign had an observed rate of at least 9%. Another interesting pattern is the degree of variation between the observed rate and the approximated rate. For example, Symantec shows the greatest difference with

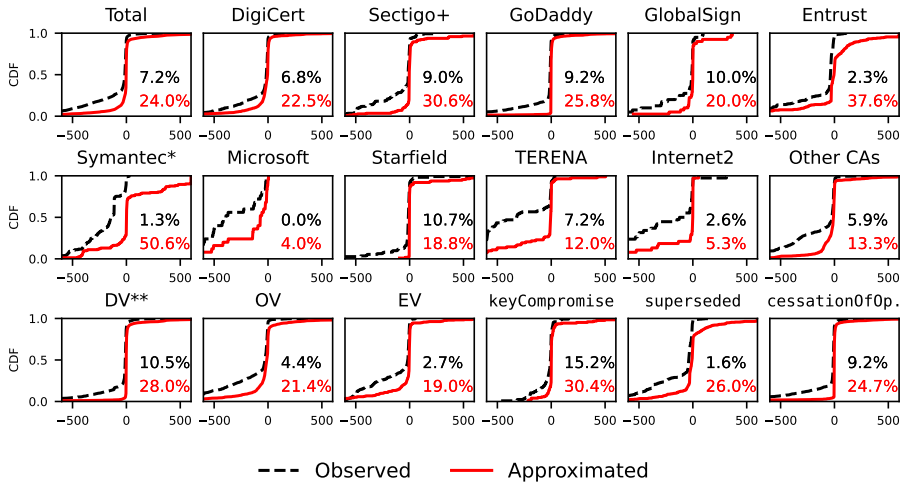


Fig. 14. Individual CDFs of lifetime minus (−) the revocation time. This corresponds to placing the revocation instance to take place at $x = 0$, meaning that all certificates where $x \geq 1$ were used for more than a day after being revoked. The exact proportions are represented by the percentages displayed in each graph. In addition to per-CA and per-type breakdowns, we also include CDFs for three noteworthy revocation reasons.

the next to lowest observed rate overall at 1.3%, but the highest approximated rate at 50.6%. In this case, the observed rate is inaccurate due to the fact that most of the Symantec issued certificates were observed just once.

As earlier theorized, if involuntary revocations are the contributing factor to certificates being used despite being revoked, then Microsoft is worth looking into as it has the lowest observed and approximated rate of all included CAs. One thing to note is that Microsoft does not issue certificates to the public[3], meaning one could say that they technically do not issue third-party certificates. While looking at Microsoft's policies regarding revocations, it becomes apparent that only they themselves (per their policy, as they solely issue certificates to affiliated domains) can request a revocation, indicating that involuntary revocations should not happen. This would furthermore strengthen the hypothesis that certificates that were used after revocation were also involuntarily revoked.

Late post-usage of certificates with potentially compromised key: The revocation category with the highest observed rate is, worryingly enough, the case of key compromises. The vast majority of these were issued by DigiCert and their intermediates such as Encryption Everywhere and RapidSSL. Encryption Everywhere stands out as they are responsible for the biggest share of key compromises (mostly wildcard DV certificates) while also offering a service where third parties (primarily hosting providers) can upsell and distribute DigiCert certificates [13].

[3] From our data, Microsoft has only issued certificates to themselves or affiliated organizations, such as MSN or Skype etc.

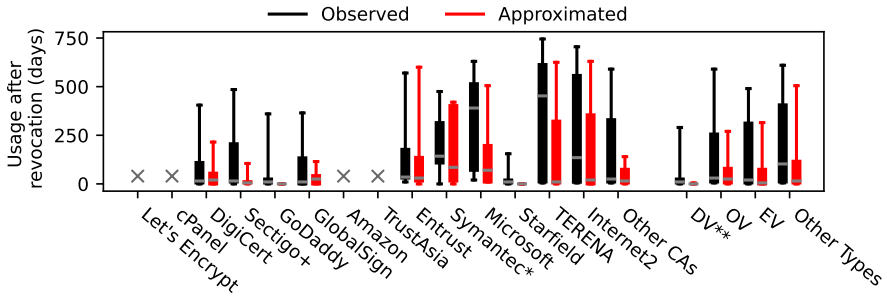


Fig. 15. Days of usage after revocation measured by observed/approximated lifetime per CA and type. The × implies that no revocation times are available.

Time certificates were advertised after being revoked: As shown in Figure 15, the exact amount of time certificates were advertised after being revoked varies greatly depending on the issuing CA. In the aggregate, the observed lifetime measurement displays a longer period of revocation usage compared to the approximated lifetime. This once again highlights that the approximated lifetime is a discernibly conservative measurement and suggests that the actual lifetime is often greater than the approximations. One surprising pattern is seen for Microsoft, Terena, and Internet2 which have some of the lowest overall revocation rates, yet some of the longest actual revocation usage. When looking at the validation types, it is clear that DV (10 days) has a significantly shorter median usage period compared to OV (30 days) and EV (20 days), which could simply be due to the, in general, shorter validity periods.

4.8 Key takeaways

In this section, we have characterized and analyzed certificate replacements and corresponding revocations. We see big differences in CAs' revocation rates (0–17%) and statistical testing of the data shows big differences in retention rates, however seemingly unrelated to revocation ratios.

For CAs with variable validity periods, we see that revoked certificates typically had significantly longer validity periods than non-revoked. Naturally, the former show a significantly higher degree of overlap compared to the latter and the data further shows that revocations often occurring early on in a certificate's lifetime. The per-CA analysis found that for CAs more prominently using automated solutions, there is only a very limited difference in overlap between revoked and non-revoked certificates. This suggests a risk of mismanagement in cases of automation. Non-revoked certificates exhibit a higher frequency of gaps, while the (few) cases of revocations with gaps tend to result in larger gaps.

The revocation time analysis reveals both cases of certificates being revoked after their lifetime and those being used after being revoked. Our approximations suggests that 24% of revoked certificates are used after revocation. Worryingly, key compromises are the most common category observed in use after revocation.

5 The revocation problem

5.1 Observed problem: Certificate (mis)management

Our results suggest that subscribers are mismanaging revoked certificates. While we believe that the direct consequences of using these revoked certificates are benign in most circumstances, it sets a bad standard and puts users at risk. This section looks closer at the certificate management problem of revoked certificates.

Certificates used despite being revoked: We have found the usage of revoked certificates to be a relatively widespread phenomena with a total average of 7% (observed) or 24% (approximated) of revoked certificates being used after the revocation. One explanation could be that subscribers revoke certificates but do not immediately replace them, meaning that certificates will be advertised despite being revoked. Perhaps the subscriber is lacking competence, or it could simply be that no replacement certificates have been prepared beforehand (and getting one issued is a slow process—especially for EV certificates). Although the idea of subscribers revoking their own certificates without preparing replacements first might seem strange, it is reasonable in cases of, for instance, key compromises where swift action is necessary.

Another, perhaps more probable, explanation is involuntary revocations. Since almost all CAs reserve the right to revoke certificates for different reasons without the subscribers' consent, this means that subscribers might not even be aware of these events. Even when subscribers are made aware of such events, the same issues with lacking competence and/or lead time might prevent subscribers from swiftly replacing certificates even if the CA tries to facilitate the process. One possible way to increase transparency and help improve best practices is to complement the revocation reasons included in CRLs with information about who requested the revocation.

Used well after revocation: Thus far, we have discovered that usage after revocation is not a rare (however unwanted) event for revoked certificates. Assuming most of them were involuntarily revoked could explain, but not justify, the certificates being advertised for a few days, since it can take a while to obtain a replacement. The problem becomes clear when considering that the majority of certificates approximated to be used after revocation were advertised for more than a week (53%) and many for more than a month (33%).

Comparing with gaps and overlaps: An interesting parallel can be drawn with the gaps in validity time found in the non-revoked CRs since serving an expired certificate should be comparable with serving a revoked one: both are examples of mismanagement. Gaps are not as frequent a phenomenon as usage after revocation but still relatively common (especially if disregarding Let's Encrypt and cPanel with their automated replacement services). Comparing Figure 10 and 15 we see that gaps typically only last a day or two while revocation usage is much longer on average. We believe that the primary reason for this discrepancy is the fact that for the former case, all major web browsers display warnings informing the user that the domain in question cannot be trusted if it adver-

tises an expired certificate [1]. This motivates subscriber to replace the expired certificate since they in all likelihood wish their website to appear trustworthy.

The same cannot be said for revoked certificates, even though—similar to an expired certificate—a revoked certificate should also be considered invalid (untrustworthy). The difference comes down to the fact that many browsers, including Chrome, Edge and almost all mobile browsers, to a large extent do not check the revocation status of a certificate at all [29,31,32,43]. As a result, most end-users (Chrome and Edge together accounting for about 70% of the global browser market share [41,42]) trying to access a website with a revoked certificate will receive no warning, leaving the user oblivious to the fact that they were served a non-trustable certificate, and leading to the subscriber not being aware/not caring (compared to expiry) that their certificate is revoked.

In contrast to gaps and uses after revocation, overlaps signal a better certificate management as an increased overlap in the case of revocations implies that at least some effort has been given to introduce a replacing certificate earlier than in the case of non-revocation. Consequently, examples like the average management of certificates from Let's Encrypt and Sectigo where the differences in overlap between revoked and non-revoked certificates are not that prominent instead implies a mismanagement of certificate replacements.

The solution at brief: Given the mismanagement that we see in our data, we argue that there is need for improvement in terms of automation regarding revocation events. We discuss a possible solution in Section 5.3.

5.2 Evolution of revocation protocols and current trends

In addition to mismanagement of revocations, it is evident from our findings that revocations are not being respected. This can largely be attributed to CRL and OCSP not being well suited for the internet in their current forms. To provide some context before we suggest how to address the certificate replacement problem associated with revoked certificates, we next discuss some related trends and on-going improvements to the revocation protocols themselves.

Challenges with current revocation protocols: It might seem strange that browser vendors would choose to disregard revocation checking when, for instance, a compromised certificate can have quite dire consequences. The reality of the situation, however, is that both CRLs and OCSP have flaws that make them less than ideal for the modern internet landscape [25]. First, both protocols add delay when making HTTPS requests (especially CRLs since they can be several MBs in size), and since speed is a big selling point for web browsers, this becomes a rather undesirable trait. Second, each of the two has a single point of failure, meaning that if a CRL distribution point or an OCSP server becomes unavailable, a large number of certificates will be affected. Finally, CRLs are usually updated in set time intervals (e.g., every 24 hours) meaning the protocol can be quite ineffective or slow against certificate compromises, and OCSP, that is designed to improve upon some of the weak points of CRLs (less overhead and always up-to-date revocation statuses), has privacy concerns since CAs are able

to monitor browsing habits by checking which domains have their revocation statuses requested (by domain visitors contacting their OCSP responders).

Revocation protocol improvements: One of the most widely deployed revocation protocol improvements as of today is OCSP stapling, which tackles some of the inherent flaws of the regular OCSP by being push-based instead of pull-based [12]. This means that the client (browser) no longer has to request the revocation status of a certificate as it will automatically be fetched and presented by the server, leading to less overhead and fewer privacy concerns for the user. OCSP stapling still has limitations, however, as the server will not pre-fetch any intermediate certificates in a certification chain, unless the "multi-stapling" extension is enabled [34], leading to additional OCSP server requests regardless. More importantly, clients will usually accept a certificate as valid if they are unable to verify its revocation status via CRL/OCSP, also known as a "soft-fail". The consequence of this is that a potential MITM attacker, possessing a compromised (and revoked) certificate, could simply intercept any outgoing OCSP requests or incoming staples, effectively forcing the client to accept the revoked certificate as valid. To combat this significant vulnerability, the OCSP Must-Staple extension was added, which if enabled in a certificate requires an OCSP staple to be included in the TLS handshake or the connection will be terminated, also known as a "hard-failure". However, during a 2018 study Chung et al. [12] found that only a very small fraction of certificates supported the extension and that none of the major browsers except Mozilla checked if the OCSP staple was included. Other recent studies have confirmed that Firefox supports OCSP Must-Staple, but Chrome does not [24].

Today, most major browsers use a proprietary push-based protocol for revocations. However, such sets typically only cover a very small fraction of all revocations on the web. Improved coverage has been achieved by using a CRL/OCSP aggregator that collects certificates from CT logs and IP scans and then searching through available CRLs and making status requests from OCSP responders [25]. By using various filters and compression techniques, CRLite [25] reduces the size of the complete list of active revocation statuses to a ∼10 MB download, with daily ∼0.5 MB updates to keep revocations fresh. Mozilla incorporated CRLite into their nightly build of Firefox in 2020, replacing OCSP statuses [21]. With the mandatory use of CT logs, Mozilla is able to rely solely on these (without IP scans), reducing the revocation delay to six hours. Other similar solutions include Chrome's CRLSets [43] and Firefox's OneCRL [32].

The rebirth of CRL: After having declined steadily since 2015 [16], CRL has regained new interest from CAs and browsers with the introduction of solutions like CRLite and CRLSets. In Sept. 2022, Let's Encrypt announced that they would begin supporting CRLs to be able to facilitate the on-going transition of browsers adopting browser-summarized CRLs [17]. In July 2023, the CA/Browser Forum decided to require CRLs (from previously being optional) and instead making OCSP optional [8]. This means that the tables have turned in favor of CRL leaving OCSP behind – at least for now.

Trend towards shorter validity periods: Certificates with long validity periods pose a greater security risk in general compared to short-lived ones. If, for instance, a private key is compromised during the first month of a one-year certificate, clients will be vulnerable to MITM attacks for the remaining 11 months while a 90-day certificate would only leave them exposed for 2 months.

The concept of short-lived certificates has been widely discussed and advocated for [39]. This is also evident in recommendations by the CA/Browser Forum, which have shifted from stipulating a maximum validity of three years to one year. Furthermore, similar to when Apple announced that they would only accept certificates with a validity time of 398 days (as compared with the then-standard 825 days) [2], Google in 2023 announced that they intended to propose a reduction of certificate validity from 398 days to 90 days [45]. Even though this initiative is yet to be taken in the CA/Browser Forum, the BR has since then introduced a short-lived certificate with a maximum of 10 days validity starting March 15, 2024, that will be reduced to 7 days starting March 15, 2026 [8]. We note that this trend also likely will push subscribers to use automated certificate management solutions.

5.3 Suggested solution: Automation of certificate management

The Automatic Certificate Management Environment (ACME) protocol [3] offers automation solutions for certificate issuance, verification, and revocation. However, despite its automation capabilities, some clients (e.g., EFF's Certbot for Let's Encrypt certificates) still require manual intervention for reissuance outside regular intervals [33]. As evident from our results, this can result in post-revocation usage of revoked certificates and otherwise late certificate replacements of these certificates. To address these issues, we argue that automated solutions must better incorporate reissuances when a certificate is revoked.

Notification of subscribers and automated reissuance: Our data, showcasing certificate mismanagement, as evident in validity gaps and post-revocation usage, for example, calls for enhanced automation, particularly concerning revocation events. Initiating revocation events should ideally involve the CA or the subscriber, with a preference for the CA to notify the subscriber promptly when revoking a certificate at their discretion. In the case that the CA does not provide such notification, the subscriber would be dependent on the monitoring of CRLs and/or OCSP. Subscriber-initiated revocations should be accompanied by immediate reissuance requests if necessary.

Collaboration between CAs and ACME providers: Based on our findings, we recommend collaborative efforts between CAs and ACME client providers to bolster automated responses to revocation events. This collaboration aims to prevent the use of revoked certificates, enhance the robustness of revocation handling, and ultimately improve the secure availability of websites online while safeguarding users.

Enhancing certificate management through automation and collaboration is pivotal for addressing the challenges associated with certificate mismanagement.

By automating processes, encouraging shorter validity periods, and strengthening responses to revocation events, we can mitigate risks, bolster security, and ensure the trustworthiness of web communications.

6 Related works

Certificate replacements: Previous studies on certificate replacements are relatively scarce with the more prominent ones focusing on mass revocation events like Heartbleed. Heartbleed was discovered in 2014 and is a bug found in an older version of OpenSSL which made it possible to extract sensitive data from the affected servers [14]. This prompted several studies to be conducted on certificates during this event, including the works by Zhang et al. [48] who analyzed certificate management by looking at reissues (replacements) and revocations, and Liu et al. [29] who focused only on revocations and CRL/OCSP characteristics. Omolola et al. found that at least 28% of domains affected by the Let's Encrypt mass revocation and with a history of regular reissuance managed to reissue their certificates within a week [33]. Perhaps most closely to ours is the work by Bruhner et al. [5]. In this paper, we make use of a specific subset of the certificate replacement relationships that they identified, extracted, and analyzed. In contrast to them, and other prior work, we focus on the relative comparison of certificate replacement relationships (e.g., gaps, overlaps, etc.) of revoked vs. non-revoked certificates. This is achieved by augmenting the replacement set with the revocation data of Korzhitskii et al. [23], enabling us to analyze the intersection of observed certificates and announced revocations. A recent work by Ma et al. [30] looks at certificate *invalidations* during a certificate's lifetime. This includes certain revocations (key compromises) but also changes in domain registrant or managed TLS certificates that do not necessarily trigger revocations, even if resulting in stale certificates. Furthermore, Ma et al. focus on certificate invalidations, irrespective of whether the certificate is actively in use or replaced. In contrast, our work relies solely on certificates observed in use with corresponding revocation status data available.

Network scans: The certificate replacements used here were all based on data from Rapid7's network scans. Before the launch of ZMap [15] (the scanner used by Rapid7), collecting data on TLS certificates was a more tedious and rather resource-intensive task. Nevertheless, comprehensive studies were made despite this. For example, Holz et al. [20] conducted longitudinal passive and active scans on popular HTTPS domains at the time and found several causes for concern in the Web PKI landscape, particularly in broken certification chains and subject names. Durumeric et al. conducted a similar study using ZMap, scanning the IPv4 address space on port 443 (HTTPS) for over a year and retrieving roughly 42 million unique certificates. Their statistics included issuing CAs, validity periods, encryption types, and revocation reasons.

Revocations: There has been plenty of works in other areas of certificate revocations as well. Zhu et al. [49] looked at the performance of the OCSP protocol in practice and found improvements in latency and wider deployment

compared to a few years prior. Kim et al. [22] studied the revocation process itself and found a handful of security threats such as CAs being slow to revoke, inaccurate "effective" revocation dates, and missing/unavailable CRL distribution points or OCSP servers. Others have studied how the status of revoked certificates sometimes change after the certificate have expired [23] or focused on the relative timing and reason of revocations [19]. However, neither of these studies considered the replacing certificate.

Reduced validity time: As discussed in Section 5.2, there is a push for short-lived certificates. Motivated by observations from their study of certificate replacements, Bruhner et al. [5] introduced the concept of "parent and child" certificates that allow for key re-usage, allowing them to use one-week certificates without introducing substantial overhead to the existing PKI. Further motivation was provided by the aforementioned study by Ma et al. [30], which estimated that a reduction of validity time of certificate from the current 398 days to (the now forthcoming) 90 days would be able to reduce the overall prevalence of staleness with over 75%, also noting that automation is a double-edge sword both enabling further reductions in certificate lifetime but at the same time presenting a risk of automatic issuance of soon-to-be stale certificates. However, in general, these trends motivate the need for good automation solutions that can provide timely reissuance both during normal conditions and when certificates are revoked. As highlighted by our results, current automation solutions do not yet appear to be satisfactory in the case that a certificate is revoked.

7 Conclusion

This paper presents the first comprehensive comparative characterization of certificate replacements of revoked certificates. Our study uncovers the complexities of certificate management in web security, delivering critical contributions and insights, including the effects revocations have on replacement behavior. Our analysis revealed significant disparities in certificate management practices among Certificate Authorities (CAs), with varying revocation rates and retention rates, demonstrating the need for standardized practices. While we have showed that certificate replacements help prevent post-revocation usage by resulting in notably longer overlaps, we alarmingly find a substantial post-revocation usage of certificates, with 7% directly observed and an estimated 24% of cases. This raises questions about the effectiveness of current revocation mechanisms and the effectiveness of replacements. To address these challenges, we advocate for enhanced automation in managing revocation events. We propose a collaborative approach between CAs and ACME client providers, emphasizing proactive notification and immediate reissuance upon revocation to bolster web security. In summary, our research offers a comprehensive understanding of certificate replacement dynamics and calls for automation and cooperation to reduce risks, strengthen security, and maintain trust in web communications.

Acknowledgments. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

1. Akhawe, D., Felt, A.P.: Alice in Warningland: A Large-Scale Field Study of Browser Security Warning Effectiveness. In: Proceedings of the USENIX Security Symposium. pp. 257–272. USENIX Security '13, USENIX Association, Washington, D.C. (2013), https://www.usenix.org/conference/usenixsecurity13/technical-sessions/presentation/akhawe
2. Apple: About upcoming limits on trusted certificates (2020), https://support.apple.com/en-us/102028
3. Barnes, R., Hoffman-Andrews, J., McCarney, D., Kasten, J.: Automatic Certificate Management Environment (ACME). RFC 8555 (2019). https://doi.org/10.17487/RFC8555
4. Boeyen, S., Santesson, S., Polk, T., Housley, R., Farrell, S., Cooper, D.: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. RFC 5280 (2008). https://doi.org/10.17487/RFC5280
5. Bruhner, C.M., Linnarsson, O., Nemec, M., Arlitt, M., Carlsson, N.: Changing of the Guards: Certificate and Public Key Management on the Internet. In: Proceeding of Passive and Active Measurement Conference. pp. 50–80. PAM '22, Virtual (2022). https://doi.org/10.1007/978-3-030-98785-5_3
6. CA/Browser Forum: Ballot SC31: Browser Alignment (2020), https://cabforum.org/2020/07/16/ballot-sc31-browser-alignment/
7. CA/Browser Forum: Guidelines for the Issuance and Management of Extended Validation Certificates (2022), https://cabforum.org/wp-content/uploads/CA-Browser-Forum-EV-Guidelines-1.8.0.pdf
8. CA/Browser Forum: Ballot SC-063 v4: Make OCSP Optional, Require CRLs, and Incentivize Automation (2023), https://cabforum.org/2023/07/14/ballot-sc-063-v4make-ocsp-optional-require-crls-and-incentivize-automation/
9. CA/Browser Forum: Baseline Requirements for the Issuance and Management of Publicly-Trusted Certificates Version 2.0.1 (2023), https://cabforum.org/wp-content/uploads/CA-Browser-Forum-BR-v2.0.1.pdf
10. Certificate Transparency: Our Successes, https://certificate.transparency.dev/community/#successes-grid
11. Chung, T., Liu, Y., Choffnes, D., Levin, D., Maggs, B.M., Mislove, A., Wilson, C.: Measuring and Applying Invalid SSL Certificates: The Silent Majority. In: Proceedings of the Internet Measurement Conference. pp. 527–541. IMC '16, ACM, Santa Monica, CA (2016). https://doi.org/10.1145/2987443.2987454
12. Chung, T., Lok, J., Chandrasekaran, B., Choffnes, D., Levin, D., Maggs, B.M., Mislove, A., Rula, J., Sullivan, N., Wilson, C.: Is the Web Ready for OCSP Must-Staple? In: Proceedings of the Internet Measurement Conference. pp. 105–118. IMC '18, ACM, Boston, MA (2018). https://doi.org/10.1145/3278532.3278543
13. DigiCert: DigiCert Encryption Everywhere Partner Program (2020), https://www.digicert.com/content/dam/digicert/pdfs/guide/partner-program-guide-en.pdf
14. Durumeric, Z., Li, F., Kasten, J., Amann, J., Beekman, J., Payer, M., Weaver, N., Adrian, D., Paxson, V., Bailey, M., Halderman, J.A.: The Matter of Heartbleed. In: Proceedings of the Internet Measurement Conference. pp. 475–488. IMC '14, ACM, Vancouver, BC, Canada (2014). https://doi.org/10.1145/2663716.2663755

15. Durumeric, Z., Wustrow, E., Halderman, J.A.: ZMap: Fast Internet-wide Scanning and Its Security Applications. In: Proceedings of the USENIX Security Symposium. pp. 605–620. USENIX Security '13, USENIX Association, Washington, D.C. (2013), https://www.usenix.org/conference/usenixsecurity13/technical-sessions/paper/durumeric
16. Farhan, S.M., Chung, T.: Exploring the Evolution of TLS Certificates. In: Proceeding of Passive and Active Measurement Conference. pp. 71–84. PAM '23, Virtual (2023). https://doi.org/10.1007/978-3-031-28486-1_4
17. Gable, A.: A New Life for Certificate Revocation Lists - Let's Encrypt (2022), https://letsencrypt.org/2022/09/07/new-life-for-crls.html
18. Google Security Blog: Chrome's Plan to Distrust Symantec Certificates (2018), https://security.googleblog.com/2017/09/chromes-plan-to-distrust-symantec.html
19. Halim, A., Danielsson, M., Arlitt, M., Carlsson, N.: Temporal Analysis of X.509 Revocations and their Statuses. In: 2022 IEEE European Symposium on Security and Privacy Workshops. pp. 258–265. EuroS&PW '22, Genoa, Italy (2022). https://doi.org/10.1109/EuroSPW55150.2022.00032
20. Holz, R., Braun, L., Kammenhuber, N., Carle, G.: The SSL Landscape: A Thorough Analysis of the x.509 PKI Using Active and Passive Measurements. In: Proceedings of the Internet Measurement Conference. pp. 427–444. IMC '11, ACM, Berlin, Germany (2011). https://doi.org/10.1145/2068816.2068856
21. Jones, J.: Design of the CRLite Infrastructure (2020), https://blog.mozilla.org/security/2020/12/01/crlite-part-4-infrastructure-design/
22. Kim, D., Kwon, B.J., Kozák, K., Gates, C., Dumitraş, T.: The Broken Shield: Measuring Revocation Effectiveness in the Windows Code-Signing PKI. In: Proceedings of the USENIX Security Symposium. pp. 851–868. USENIX Security '18, USENIX Association, Baltimore, MD (2018), https://www.usenix.org/conference/usenixsecurity18/presentation/kim
23. Korzhitskii, N., Carlsson, N.: Revocation Statuses on the Internet. In: Proceeding of Passive and Active Measurement Conference. pp. 175–191. PAM '21, Virtual (2021). https://doi.org/10.1007/978-3-030-72582-2_11
24. Larisch, J., Aqeel, W., Lum, M., Goldschlag, Y., Kannan, L., Torshizi, K., Wang, Y., Chung, T., Levin, D., Maggs, B.M., Mislove, A., Parno, B., Wilson, C.: Hammurabi: A Framework for Pluggable, Logic-Based X.509 Certificate Validation Policies. In: Proceedings of the Conference on Computer and Communications Security. pp. 1857–1870. CCS '22, ACM, Los Angeles, CA (2022). https://doi.org/10.1145/3548606.3560594
25. Larisch, J., Choffnes, D., Levin, D., Maggs, B.M., Mislove, A., Wilson, C.: CRLite: A Scalable System for Pushing All TLS Revocations to All Browsers. In: 2017 IEEE Symposium on Security and Privacy. pp. 539–556. S&P '17, IEEE, San Jose, CA (2017). https://doi.org/10.1109/SP.2017.17
26. Laurie, B., Langley, A., Kasper, E., Messeri, E., Stradling, R.: Certificate Transparency Version 2.0. RFC 9162 (2021). https://doi.org/10.17487/RFC9162
27. Let's Encrypt: Integration Guide. Internet Security Research Group (2016), https://letsencrypt.org/docs/integration-guide/
28. Let's Encrypt: 2020.02.29 CAA Rechecking Bug (2020), https://community.letsencrypt.org/t/2020-02-29-caa-rechecking-bug/114591
29. Liu, Y., Tome, W., Zhang, L., Choffnes, D., Levin, D., Maggs, B., Mislove, A., Schulman, A., Wilson, C.: An End-to-End Measurement of Certificate Revocation in the Web's PKI. In: Proceedings of the Internet Measurement Conference. pp.

183–196. IMC '15, ACM, Tokyo, Japan (2015). https://doi.org/10.1145/2815675.2815685
30. Ma, Z., Faulkenberry, A., Papastergiou, T., Durumeric, Z., Bailey, M.D., Keromytis, A.D., Monrose, F., Antonakakis, M.: Stale TLS Certificates: Investigating Precarious Third-Party Access to Valid TLS Keys. In: Proceedings of the Internet Measurement Conference. pp. 222–235. IMC '23, ACM, Montreal QC, Canada (2023). https://doi.org/10.1145/3618257.3624802
31. Microsoft: Microsoft Edge - Policies (2023), https://learn.microsoft.com/en-us/deployedge/microsoft-edge-policies
32. Mozilla: CA/Revocation Checking in Firefox (2021), https://wiki.mozilla.org/CA/Revocation_Checking_in_Firefox#OneCRL
33. Omolola, O., Roberts, R., Ashiq, M.I., Chung, T., Levin, D., Mislove, A.: Measurement and Analysis of Automated Certificate Reissuance. In: Proceeding of Passive and Active Measurement Conference. pp. 161–174. PAM '21, Virtual (2021). https://doi.org/10.1007/978-3-030-72582-2_10
34. Pettersen, Y.N.: The Transport Layer Security (TLS) Multiple Certificate Status Request Extension. RFC 6961 (2013). https://doi.org/10.17487/RFC6961
35. Rapid7: Open Data: SSL Certificates, https://opendata.rapid7.com/sonar.ssl/
36. Rapid7: Project Sonar, https://www.rapid7.com/research/project-sonar/
37. Santesson, S., Myers, M., Ankney, R., Malpani, A., Galperin, S., Adams, D.C.: X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP. RFC 6960 (2013). https://doi.org/10.17487/RFC6960
38. Sectigo: What is a Self-Signed Certificate (2021), https://sectigo.com/resource-library/what-is-a-self-signed-certificate
39. Sheffer, Y., Lopez, D., de Dios, O.G., Pastor, A., Fossati, T.: Support for Short-Term, Automatically Renewed (STAR) Certificates in the Automated Certificate Management Environment (ACME). RFC 8739 (2020). https://doi.org/10.17487/RFC8739
40. SSLmate: Certificate Transparency Log Growth, https://sslmate.com/labs/ct_growth/
41. Statcounter GlobalStats: Browser Market Share Worldwide (2023), https://gs.statcounter.com/browser-market-share
42. Statista: Global market share held by leading internet browsers from January 2012 to January 2023 (2023), https://www.statista.com/statistics/268254/market-share-of-internet-browsers-worldwide-since-2009/
43. The Chromium Projects: CRLSets, https://www.chromium.org/Home/chromium-security/crlsets/
44. The Chromium Projects: Chrome Root Program Policy, Version 1.4 (2023), https://www.chromium.org/Home/chromium-security/root-ca-policy/
45. The Chromium Projects: Moving Forward, Together (2023), https://www.chromium.org/Home/chromium-security/root-ca-policy/moving-forward-together/
46. TrustAsia: TrustAsia CA Certificate Practice Statement (CPS) V1.1 (8 2020), https://repository.trustasia.com/repo/cps/TrustAsia-Global-CP-CPS_EN_V1.1.pdf
47. Yee, P.E.: Updates to the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. RFC 6818 (2013). https://doi.org/10.17487/RFC6818
48. Zhang, L., Choffnes, D., Dumitraş, T., Levin, D., Mislove, A., Schulman, A., Wilson, C.: Analysis of SSL Certificate Reissues and Revocations in the Wake of

Heartbleed. Communications of the ACM **61**(3), 109–116 (2018). https://doi.org/10.1145/3176244

49. Zhu, L., Amann, J., Heidemann, J.: Measuring the Latency and Pervasiveness of TLS Certificate Revocation. In: Proceeding of Passive and Active Measurement Conference. pp. 16–29. PAM '16, Heraklion, Greece (2016). https://doi.org/10.1007/978-3-319-30505-9_2